REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 24, 2001	Final POrogress

4. TITLE AND SUBTITLE
Electro-Thermal Behavioral Modeling

5. FUNDING NUMBERS
N68171-99-M-5800
R&D 8286-EE-01

6. AUTHOR(S)
Michael Steer†, Chris Snowden, Frank Hart†, Daniel Stephenson† and Robert Johnson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
School of Electrical and Electronic Engineering
University of Leeds
Leeds, LS2 9Jt
United Kingdom

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Army Research Laboratory - European Research Office
223 Old Marylebone Road
London, NW1 5TH
England

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12 a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This project developed the technology required to develop electro-thermal behavioral models of microwave circuits and subsystems. Electrical behavioral models themselves are poorly developed however we have developed an electrical behavioral model that can be readily extracted from single tone measurements or simulations and capture baseband effects and in particular thermal effects. The core technology is based on mathematics developed by the authors called the Arithmetic Operator Method which enables the basic mathematical operations such as addition, multiply, divide and subtraction of signals to be performed by operating on their spectra. An example of application of the method to the modeling of a logarithmic amplifier are presented. The model was implemented in MATLAB code and this code is available by contacting m.b.steer@ieee.org. The code has been transferred to two MURIs (Multidsciplinary University Research Initiatives) managed by the Army Research Office. In the first MURI the work developed here will be used to model quasi-optical power combining amplifiers. In the second MURI the work will be used to model the co-site interference problem.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Microwave System; Behavioral Modeling; Electro-Thermal	
	16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT	18. SECURITY CLASSIFICATION ON THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev.2-89)
Prescribed by ANSI Std. 239-18
298-102

Electro-Thermal Behavioral Modeling

Michael Steer[†‡] ,Christopher Snowden[†] ,Frank Hart[‡], Daniel Stephenson[‡] and Robert Johnson[†]

[†]University of Leeds, School of Electrical and Electronic Engineering
[‡]North Carolina State University, Department of Electrical and Computer Engineering
Contact m.b.steer@ieee.org for copies of this report and for code.

Abstract

This project developed the technology required to develop electro-thermal behavioral models of microwave circuits and subsystems. Electrical behavioral models themselves are poorly developed however we have developed an electrical behavioral model that can be readily extracted from single tone measurements or simulations and capture baseband effects and in particular thermal effects. The core technology is based on mathematics developed by the authors called the Arithmetic Operator Method which enables the basic mathematical operations such as addition, multiply, divide and subtraction of signals to be performed by operating on their spectra. An example of application of the method to the modeling of a logarithmic amplifier are presented. The model was implemented in MATLAB code and this code is available by contacting m.b.steer@ieee.org. The code has been transferred to two MURIs (Multidsciplinary University Research Initiatives) managed by the Army Research Office. In the first MURI the work developed here will be used to model quasi-optical power combining amplifiers. In the second MURI the work will be used to model the co-site interference problem.

Table of Contents

1. Introduction

The Arithmetic Operator Method (AOM) was pioneered by Chang and Steer and has previously been described in two publications, [1] and [2]. It has been successfully applied to electronic circuit behavioral modeling by Gard et al [3] and by de Carvalho and Pedro [4], [5]. This project further developed this behavioral modeling approach and implemented it in MATLAB together with a worked example and comparison to SPICE results.

At the most basic level, the Arithmetic Operator Method (AOM) is a computer-aided method for simulating the time-domain multiplication of two signals in the frequency domain, using convolution in the frequency domain. However, its applications are not limited to the simple multiplication of two time-domain signals that one would find in a typical radio-frequency mixer, as a case study will later illustrate. We will see that AOM can be used to model transfer functions of considerable complexity, as long as three prerequisites are met:

1. The transfer function to be modeled can be expressed as a truncated infinite series form and must be a mathematically "analytic" function as defined by Taylor and Mann [6];

2. The spectral content of the input is known; and

3. The domain of the spectral content (i.e. frequencies, but not amplitude or phase) of the output is either known or pre-specified.

At first glance, the last prerequisite might seem difficult to meet. However, we will see that when the input is limited to a finite number of sinusoids of arbitrary frequency, then it is possible to specify the domain of the spectral content of the output. In the case study, we demonstrate how AOM can be used to model the response of a logarithmic amplifier to a two-tone sinusoidal input signal.

2. Mathematical Description

2. 1 Mathematical Foundation

The theoretical background underpinning the Arithmetic Operator Method was first described by Oppenheim and Schafer [7] in 1975. In chapter 10 of their book, *Digital Signal Processing,* the authors describe a "Generalized Superposition" principle that has extended utility beyond the usual class of Linear, Time-Invariant (LTI) systems. The essential properties of such systems is that their operators must be both commutative and associative over the domains on which they operate. Boyce and DiPrima [8] demonstrated that not all functions which can be convolved are associative, and this puts some restrictions on the form of the functions. The functions we consider will meet those restrictions. Such systems can be modeled using a linear transformation on a vector space, i.e. using matrix algebra. Systems that perform multiplication in the time domain, which corresponds to convolution in the frequency domain, exhibit the properties required for "generalized superposition" – even though they are inherently non-linear. The resulting linear transformation implements the essence of the Arithmetic Operator Method. We present here the development of the transformation.

2.2 Time and Frequency Domain Descriptions of Signals of Interest

Let's begin with two time-domain signals, $x(t)$ and $z(t)$, both of which are composed of a finite number of sinusoids of arbitrary frequency content. For the purposes of the development here, we will assume that x and z have spectral content at the same frequencies, although with different amplitudes and phases. This is not restrictive, since if x and z were composed of sinusoids of different frequencies, we could describe

both using a common (i.e. union) set of frequencies with some amplitude coefficients in x and z both set to 0. Thus, the development that follows here for $x(t)$ will also apply to $z(t)$. We can express $x(t)$ as the sum of a set of sinusoidal signals as follows:

$$x(t) = \sum_{n=0}^{N} x_n(t) = \sum_{n=0}^{N} |x_n| \cos(\omega_n t + \phi_n) = \sum_{n=0}^{N} |x_n| \frac{e^{j(\omega_n t + \phi_n)} + e^{-j(\omega_n t + \phi_n)}}{2} \tag{1.1}$$

Here, we have expressed each sinusoid using the cosine function with frequency $\omega_n = 2\pi f_n$, amplitude $|x_n|$, and phase ϕ_n. Following that, we have used Euler's equation $(e^{j\theta} = \cos\theta + j\sin\theta)$ to express the cosine function as the sum of two complex exponentials.[1] We can further refine this result as follows:

$$\begin{aligned} \sum_{n=0}^{N} |x_n| \frac{e^{j(\omega_n t + \phi_n)} + e^{-j(\omega_n t + \phi_n)}}{2} &= \tfrac{1}{2} \sum_{n=0}^{N} |x_n| e^{j(\omega_n t + \phi_n)} + \tfrac{1}{2} \sum_{n=0}^{N} |x_n| e^{-j(\omega_n t + \phi_n)} \\ &= \tfrac{1}{2} \sum_{n=0}^{N} |x_n| e^{j\phi_n} e^{j\omega_n t} + \tfrac{1}{2} \sum_{n=0}^{N} |x_n| e^{-j\phi_n} e^{-j\omega_n t} \\ &= \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} + jX_{ni}] e^{j\omega_n t} + \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} - jX_{ni}] e^{-j\omega_n t} \end{aligned} \tag{1.2}$$

Here we have again used Euler's equation to express the amplitude and phase as a complex number pair, $X_{nr} \pm jX_{ni}$. If we now take the Fourier transform of $x(t)$ using the final form in equation (1.2) above, we will have:

$$\begin{aligned} X(\omega) = \mathsf{F}\{x(t)\} &= \mathsf{F}\left\{ \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} + jX_{ni}] e^{j\omega_n t} + \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} - jX_{ni}] e^{-j\omega_n t} \right\} \\ &= \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} + jX_{ni}][2\pi\delta(\omega - \omega_n)] + \tfrac{1}{2} \sum_{n=0}^{N} [X_{nr} - jX_{ni}][2\pi\delta(\omega + \omega_n)] \end{aligned} \tag{1.3}$$

Now, from the definition of the inverse Fourier transform, we have:

[1] This form will allow us to use a phasor form for the cosine function later in the development. However, it must be stressed that since we are developing a model for a non-linear system, we cannot make the usual linear phasor transformation $x(t) = \mathrm{Re}[X(\omega)]$, and then use $X(\omega)$. Our representation is exact, although we will get the benefit of phasor arithmetic through our formulation.

$$
\begin{aligned}
x(t) &= \mathsf{F}^{-1}\{X(\omega)\} \\
&= \tfrac{1}{2\pi}\int_{-\infty}^{+\infty} X(\omega)e^{j\omega t}d\omega \\
&= \tfrac{1}{2\pi}\int_{-\infty}^{+\infty}\left\{\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}+jX_{ni}][2\pi\delta(\omega-\omega_n)]+\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}-jX_{ni}][2\pi\delta(\omega+\omega_n)]\right\}e^{j\omega t}d\omega \\
&= \tfrac{1}{2\pi}(2\pi)\int_{-\infty}^{+\infty}\left\{\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}+jX_{ni}][\delta(\omega-\omega_n)]+\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}-jX_{ni}][\delta(\omega+\omega_n)]\right\}e^{j\omega t}d\omega \\
&= \int_{-\infty}^{+\infty}\left\{\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}+jX_{ni}][\delta(\omega-\omega_n)]+\tfrac{1}{2}\sum_{n=0}^{N}[X_{nr}-jX_{ni}][\delta(\omega+\omega_n)]\right\}e^{j\omega t}d\omega
\end{aligned}
\tag{1.4}
$$

Thus we see that there is a factor of 2π that appears when we follow the strict definition of the Fourier transform. This same factor is then eliminated in the inverse Fourier transform. For the purposes of our development, we will ignore this domain conversion factor as we go forward, since the ultimate result of our efforts will be a time domain signal. With this simplification, we observe that the frequency domain constants for each $\delta(\omega-\omega_n)$ component are simply $\frac{1}{2}(X_{nr}+jX_{ni})=\frac{1}{2}X_n$ for components on the right hand side of the spectrum. On the left hand side of the spectrum, each $\delta(\omega+\omega_n)$ will have a multiplicative constant of $\frac{1}{2}(X_{nr}-jX_{ni})=\frac{1}{2}(X_{nr}+jX_{ni})^*=\frac{1}{2}X_n^*$, where X_n^* denotes the complex conjugate of X_n. Of special note is what happens for the case of DC, when $n=0$. By definition, DC corresponds to the origin in the frequency domain, so there is no imaginary part to the constant X_0, i.e. $X_0=X_{0r}$. This can also be seen by going back to equation (1.2), substituting 0 for n, and working through the complex arithmetic. Note also that there is no scale factor of $\frac{1}{2}$ on the DC term. We can now rewrite $X(\omega)$ from equation (1.3), dropping the factor of 2π and taking account of the coefficient scale factors:

$$X(\omega)=\tfrac{1}{2}\sum_{n=1}^{N}X_n\delta(\omega-\omega_n)+X_0\delta(\omega)+\tfrac{1}{2}\sum_{n=1}^{N}X_n^*\delta(\omega+\omega_n) \tag{1.5}$$

To recap, the first summation is a set of frequency domain components on the right hand side of the spectrum, the second is a set of frequency domain components on the left hand side of the spectrum, and we carry the DC term separately. Now, since we will be dealing with convolution in the frequency domain shortly, it will be convenient to designate the components in the left hand side of the spectrum with negative indices, so we have:

$$X(\omega)=\tfrac{1}{2}\sum_{n=1}^{N}X_n\delta(\omega-\omega_n)+X_0\delta(\omega)+\tfrac{1}{2}\sum_{-n=1}^{N}X_{-n}^*\delta(\omega+\omega_{-n}) \tag{1.6}$$

...or, in an even more compact form:

$$X(\omega)=X_0\delta(\omega)+\tfrac{1}{2}\sum_{n=1}^{N}\{X_n\delta(\omega-\omega_n)+X_{-n}^*\delta(\omega+\omega_{-n})\} \tag{1.7}$$

Now, we can express $Z(\omega)$ similarly, i.e.:

$$Z(\omega) = Z_0\delta(\omega) + \tfrac{1}{2}\sum_{n=1}^{N}\{Z_n\delta(\omega-\omega_n) + Z_{-n}^{*}\delta(\omega+\omega_{-n})\} \tag{1.8}$$

2.3 Convolution with Reduction of the Frequency Domain to Positive Frequencies

At this point we can consider the effect of defining the time domain operation $y(t) = x(t)\cdot z(t)$. Since time domain multiplication corresponds to frequency domain convolution, we have $y(t) = x(t)\cdot z(t) \leftrightarrow Y(\omega) = X(\omega) * Z(\omega)$, so

$$\begin{aligned} Y(\omega) = &\left\{X_0\delta(\omega) + \tfrac{1}{2}\sum_{n=1}^{N}\{X_n\delta(\omega-\omega_n) + X_{-n}^{*}\delta(\omega+\omega_{-n})\}\right\} * \\ &\left\{Z_0\delta(\omega) + \tfrac{1}{2}\sum_{n=1}^{N}\{Z_n\delta(\omega-\omega_n) + Z_{-n}^{*}\delta(\omega+\omega_{-n})\}\right\} \end{aligned} \tag{1.9}$$

This is a cumbersome equation, but the effect of convolving $X(\omega)$ with $Z(\omega)$ will be a set of δ functions with frequencies ranging from $-2\omega_N$ (at summation index –2N) to $2\omega_N$ (at summation index 2N). We will thus have:

$$Y(\omega) = Y_0\delta(\omega) + \tfrac{1}{2}\sum_{n=1}^{2N}\{Y_n\delta(\omega-\omega_n) + Y_{-n}^{*}\delta(\omega+\omega_{-n})\} \tag{1.10}$$

If we examine the form of $Y(\omega)$, we see that the left hand side of the spectrum has the same amplitude and opposite phase of the right hand side of the spectrum. It thus conveys no information above and beyond that available from the right hand side of the spectrum alone, and so we can reduce the computational complexity involved in carrying out frequency domain convolution by eliminating the left hand side of the spectrum in the convolution operation. A consequence of doing this is that we must recover the amplitude and phase of each component of $y(t)$ from the coefficients of the right side of the spectrum, i.e. given that $y(t)$ has the form:

$$y(t) = \sum_{n=1}^{2N}|y_n|\cos(\omega_n t + \phi_n) + y_0 \tag{1.11}$$

Then without adjustments, we would have, for each n,

$$\tfrac{1}{2}|y_n| = \sqrt{Y_{nr}^2 + Y_{ni}^2} \tag{1.12}$$

and:

$$\phi_n = \tan^{-1}\left[\frac{Y_{ni}}{Y_{nr}}\right] \tag{1.13}$$

The phase is unaffected by our dropping of half the spectrum, but in order to recover the correct amplitude, equation (1.12) must be scaled by a factor of 2. In the matrix formulation that we are developing, we take account of this factor of two in the formulation of the matrix, leaving the multiplying vector unscaled. Since the number of convolution operations is reduced geometrically[2], while the additional operations increase linearly, this is a good tradeoff. As we proceed with the development of the method, we will show frequency domain convolution only for the right hand side of the spectrum with the expectation that the amplitude and phase of $y(t)$ are determined as above. With the spectrum limited to the right hand side, we define the spectrum for $x(t)$ from equation (1.1) as:

$$X(\omega) = X_0\delta(\omega) + \tfrac{1}{2}\sum_{n=1}^{N} X_n\delta(\omega-\omega_n) \tag{1.14}$$

Now $z(t)$ would be equivalent except for the fact that we wish to account for the output scale factor in its formulation, so we will multiply all frequency components by 2 to take this into account:

$$\begin{aligned} Z(\omega) &= Z_0\delta(\omega) + 2\tfrac{1}{2}\sum_{n=1}^{N} Z_n\delta(\omega-\omega_n) \\ &= Z_0\delta(\omega) + \sum_{n=1}^{N} Z_n\delta(\omega-\omega_n) \end{aligned} \tag{1.15}$$

Now let's consider $y(t) = x(t)\cdot z(t)$ given this redefinition in the frequency domain. We have $y(t) = x(t)\cdot z(t) \leftrightarrow Y(\omega) = X(\omega) * Z(\omega)$, so:

$$Y(\omega) = X(\omega) * Z(\omega) = \sum_{All\,\omega_k} X(\omega_k)Z(\omega-\omega_k) \tag{1.16}$$

Expanding this by substituting equations (1.14) and (1.15) into equation (1.16), we have:

[2] Consider, for example, a case where x and z are both composed of tones of 10 different frequencies. This corresponds to 21 frequency domain locations (after including DC) that must be convolved with 21 other frequencies, or computational complexity on the order of $21^2 = 441$. Restricting the frequency domain to the right hand side, we need convolve only 11 frequency domain components for a computational complexity on the order of 11^2, or 121. The ratio is just under 4:1 or 2^2:1, indicating the geometric reduction in computation.

$$Y(\omega) = X(\omega) * Z(\omega) = \sum_{All\,\omega_k} X(\omega_k) Z(\omega - \omega_k) =$$

$$\left\{X_0\delta(\omega_k) + \tfrac{1}{2}X_1\delta(\omega_k - \omega_1) + \tfrac{1}{2}X_2\delta(\omega_k - \omega_2) + \ldots \tfrac{1}{2}X_N\delta(\omega_k - \omega_N)\right\} \cdot \left| \begin{array}{l} \left\{ \begin{array}{l} Z_0\delta(\omega - \omega_k) + Z_1\delta(\omega - \omega_k - \omega_1) + Z_2\delta(\omega - \omega_k - \omega_2) + \ldots \\ + Z_N\delta(\omega - \omega_k - \omega_N) \end{array} \right\} \\ All\,\omega_k \end{array} \right. \quad (1.17)$$

$$=$$

$$\left\{X_0\delta(\omega_k) + \tfrac{1}{2}X_1\delta(\omega_k - \omega_1) + \tfrac{1}{2}X_2\delta(\omega_k - \omega_2) + \ldots \tfrac{1}{2}X_N\delta(\omega_k - \omega_N)\right\} \cdot \left| \begin{array}{l} \left\{ \begin{array}{l} Z_0\delta(\omega - \omega_k) + Z_1\delta[\omega - (\omega_k + \omega_1)] + Z_2\delta[\omega - (\omega_k + \omega_2)] + \ldots \\ + Z_N\delta[\omega - (\omega_k + \omega_N)] \end{array} \right\} \\ All\,\omega_k \end{array} \right.$$

In the equations above, we make only one stipulation – that $\omega_0 = 0$. We can now make some observations on the frequency domain convolution above. First, for a total of N distinct input tones or frequencies – all in the right hand side of the spectrum – the output $Y(\omega)$ has components at DC and 2N positive frequencies, the maximum radian frequency being $2\omega_N$. Second, exhibiting the usual sifting property of the δ function, for each distinct value of the frequency index k, only one term in $X(\omega_k)$ will be non-zero. This term will then form a set of products with all of the components of $Z(\omega - \omega_k)$, which retains the independent variable ω. Put another way, each sifted term in $X(\omega_k)$ forms product terms with all of the terms in $Z(\omega)$, but shifted in the frequency domain to the frequency dictated by the argument to the δ functions for $Z(\omega)$ called for in equation (1.17). In the course of the convolution operation, there may be a number of non-zero products for different values of the frequency index k for which the δ functions in $Z(\omega)$ happen to evaluate to the same frequency. The convolution operation calls for these terms to add linearly, conforming to the notion of generalized superposition.

To illustrate this, consider the cases of $k = 1$ and $k = 2$.

When $k = 1$, we have:

$$Y(\omega)\big|_{k=1} = \tfrac{1}{2}X_1\delta(\omega_1 - \omega_1) \cdot \left\{ \begin{array}{l} Z_0\delta(\omega - \omega_1) + Z_1\delta[\omega - (\omega_1 + \omega_1)] + \\ Z_2\delta[\omega - (\omega_1 + \omega_2)] + \ldots + \\ Z_N\delta[\omega - (\omega_1 + \omega_N)] \end{array} \right\} \quad (1.18)$$

$$= \tfrac{1}{2}X_1 \cdot \left\{ \begin{array}{l} Z_0\delta(\omega - \omega_1) + Z_1\delta(\omega - 2\omega_1) + \\ Z_2\delta[\omega - (\omega_1 + \omega_2)] + \ldots Z_N\delta[\omega - (\omega_1 + \omega_N)] \end{array} \right\}$$

And when $k = 2$, we have:

$$Y(\omega)\big|_{k=2} = \tfrac{1}{2} X_2 \delta(\omega_2 - \omega_2) \cdot \begin{Bmatrix} Z_0\delta(\omega-\omega_2)+Z_1\delta[\omega-(\omega_2+\omega_1)]+ \\ Z_2\delta[\omega-(\omega_2+\omega_2)]+\ldots+ \\ Z_N\delta[\omega-(\omega_2+\omega_N)] \end{Bmatrix} \tag{1.19}$$

$$= \tfrac{1}{2} X_2 \cdot \begin{Bmatrix} Z_0\delta(\omega-\omega_2)+Z_1\delta[\omega-(\omega_2+\omega_1)]+ \\ Z_2\delta[\omega-2\omega_2)]+\ldots Z_N\delta[\omega-(\omega_1+\omega_N)]) \end{Bmatrix}$$

Consider the frequency component $\omega_1+\omega_2$, which is indicated by any terms containing $\delta[\omega-(\omega_1+\omega_2)]$ in equations (1.18) and (1.19). From equation (1.18), there is a term at $\omega_1+\omega_2$ with the coefficient $\frac{1}{2}X_1Z_2$, while from equation (1.19), there is a term at $\omega_1+\omega_2$ with the coefficient $\frac{1}{2}X_2Z_1$. In general, the terms in each $Y(\omega)\big|_k$ will be composed of spectral components at different frequencies – each coefficient having a different argument to its δ function. This is not the most useful form to implement using computed-aided methods, as we will now see.

2.4 Formulation of a Matrix Method for the Frequency Domain Convolution

Let's examine the result of performing the complex multiplication required to create one of the components of $Y(\omega)\big|_k$. Continuing with terms that appear at the frequency $\omega_1+\omega_2$ in $Y(\omega)$, let's examine what happens when we perform the complex multiplication of X_1 and Z_2:

$$X_1Z_2 = (X_{1r}+jX_{1i})\cdot(Z_{2r}+jZ_{2i}) = \begin{aligned} & X_{1r}Z_{2r}-X_{1i}Z_{2i} \\ & +j(X_{1i}Z_{2r}+X_{1r}Z_{2i}) \end{aligned} \tag{1.20}$$

The result we obtain is a complex number composed of four product terms from the complex number pairs for X_1 and Z_2. Of particular note here is that there are real and imaginary parts of the product that will combine with other product terms occurring at $\omega_1+\omega_2$ to create real and complex parts of $Y(\omega)$. Hence this complex multiplication can be viewed as forming two real-valued linear equations. If we do this, we can formulate the product in equation (1.20) in the following matrix form:

$$X_1Z_2 = \begin{bmatrix} Y_r' \\ Y_i' \end{bmatrix} = \begin{bmatrix} X_{1r} & -X_{1i} \\ X_{1i} & X_{1r} \end{bmatrix} \begin{bmatrix} Z_{2r} \\ Z_{2i} \end{bmatrix} \tag{1.21}$$

Here we use the primes on Y merely to indicate that this product forms only a part of the output $Y(\omega)$ at the frequency $\omega_1+\omega_2$. In order to get the complete response of $Y(\omega)$ at the frequency $\omega_1+\omega_2$, it will be necessary to evaluate each $Y(\omega)\big|_k$ at the frequency $\omega_1+\omega_2$ and form the sum as suggested by the convolution summation. In other words,

$$Y(\omega_1+\omega_2) = Y(\omega_1+\omega_2)\big|_{k=1} + Y(\omega_1+\omega_2)\big|_{k=2} + \ldots + Y(\omega_1+\omega_2)\big|_{k=2N} \tag{1.22}$$

Equation (1.22) suggests extending the matrix multiplication form introduced in equation (1.21) so that a simple matrix multiplication will yield a sum of products that forms the complete response for $Y(\omega)$ at a particular frequency. As an illustrative example, consider for a moment that $Y(\omega_1+\omega_2)$ is limited to just terms in $k=1$ and $k=2$. Referring to the product terms in equations (1.18) and (1.19), then, the matrix form of equation (1.21) would be extended to add extra columns to the matrix and extra rows to the vector:

$$Y(\omega_1+\omega_2)\big|_{k=1,2} = Y(\omega_1+\omega_2)\big|_{k=1} + Y(\omega_1+\omega_2)\big|_{k=2}$$

$$= \begin{bmatrix} Y_r' \\ Y_i' \end{bmatrix}(\omega_1+\omega_2) = \begin{bmatrix} \frac{1}{2}X_{2r} & -\frac{1}{2}X_{2i} & \frac{1}{2}X_{1r} & -\frac{1}{2}X_{1i} \\ \frac{1}{2}X_{2i} & \frac{1}{2}X_{2r} & \frac{1}{2}X_{1i} & \frac{1}{2}X_{1r} \end{bmatrix} \begin{bmatrix} Z_{1r} \\ Z_{1i} \\ Z_{2r} \\ Z_{2i} \end{bmatrix} \tag{1.23}$$

This formulation can be generalized by adding two rows to the matrix for each unique output frequency in $Y(\omega)$, and it can be generalized for each additional input frequency by adding two columns to the matrix and two rows to the vector element. The matrix thus formulated has been termed the *Spectrum Transform Matrix* by Chang and Steer [2]. In matrix algebra notation, the spectrum transform matrix is denoted $\mathbf{T}_x$, the spectral vector of frequency components for $z(t)$ is denoted as $\mathbf{Z}$, and the basic operation $y(t)=x(t)\cdot z(t)$ can now be written with its frequency domain counterpart as:

$$y(t)=x(t)\cdot z(t) \leftrightarrow Y(\omega)=X(\omega)*Z(\omega)=\mathbf{T}_x\mathbf{Z} \tag{1.24}$$

2.5 Generalization of the Spectrum Transform Matrix to Arbitrary Transfer Functions

The Spectrum Transform Matrix as described in the previous section implements the frequency domain convolution for the basic time domain multiplication operation $y(t)=x(t)\cdot z(t)$. However, it is possible to further the utility of the Arithmetic Operator Method, as we will now describe. Consider $y(t)$ as an arbitrary polynomial transfer function of $x(t)$, for example

$y=a_0+a_1x+a_2x^2+\ldots+a_Mx^M=\sum_{m=0}^{M}a_mx^m$. Examining the term for m=2 more closely, we can write this is as $y(t)\big|_{m=2}=a_2x^2=a_2x\cdot x$, from which it follows that $a_2x\cdot x \leftrightarrow a_2\mathbf{T}_x\mathbf{X}$. We can then substitute the matrix product $\mathbf{T}_x\mathbf{X}$ into the term for m=3 and obtain $a_3x\cdot x^2 \leftrightarrow a_3\mathbf{T}_x(\mathbf{T}_x\mathbf{X})=a_3\mathbf{T}^2{}_x\mathbf{X}$. In general,

$$y=\sum_{m=0}^{M}a_mx^m \leftrightarrow \mathbf{Y}=\sum_{m=0}^{M}a_m\mathbf{T}_x^{M-1}\mathbf{X} \tag{1.25}$$

With this formulation, a word is in order about a convention that has been adopted into the algorithm for creating the spectrum transform matrix as described by Chang and Steer [2]. Referring back to equations (1.12) through (1.15), we see that we accounted for a lost amplitude scale factor of $\frac{1}{2}$ by multiplying the non-zero components of $Z(\omega)$ by 2, leaving a scale factor of $\frac{1}{2}$ in the Spectrum Transform Matrix. Since

$x(t)$ and $z(t)$ were both composed of sinusoids, we chose arbitrarily to scale the spectral vector up by 2. However, as we generalize the formulation of the Spectrum Transform Matrix, the spectral vector will generally remain a set of sinusoids, but the Spectrum Transform Matrix will not. In the interest of keeping the Spectral Vector description as simple as possible, we will move the $\frac{1}{2}$ scale factor from the Spectral Vector to all of the non-DC terms in the Spectrum Transform Matrix $\mathbf{T}_x$. By convention, then, the Spectral Vector $\mathbf{X}$ will always be composed of the unscaled phasor coefficients for each sinusoidal component of the input. This scale factor is termed ε in the paper by Chang and Steer [2], which also gives a detailed algorithm for construction of the Spectrum Transform Matrix.

2.6 Streamlined Method for Developing Entries in the Spectrum Transform Matrix

Now that we have seen why the Spectrum Transform Matrix construction can be useful, we need to find a streamlined way to construct it – one that is simpler than writing out all of the convolution products and evaluating terms in $Y(\omega)$ as described in equation (1.22). We will see in a moment that we can use a simple algorithm to tabulate all of the entries in the Spectrum Transform Matrix. This tabular form of the entries in the Spectrum Transform Matrix is termed the Spectrum Mapping Table by Chang and Steer [2]. It is derived by anticipating all combinations of the input tones (including intermodulation distortion products) that will result in incremental phasor components in the set of output frequencies that we have anticipated in advance. (Recall that anticipating the domain of the spectral content of the output is one of the prerequisites to using AOM.) To do this, we use a mathematical tool called the Basic Intermodulation Product Description (BIPD) table.

2.6.1 Basic Intermodulation Product Description Table

The Basic Intermodulation Product Description (BIPD) table is a list of frequencies which defines the anticipated frequency domain of the output signal $y(t)$. It is created by determining a set of frequencies of interest given the form of the input signal. Often the input signal is a non-commensurate two-tone signal, so we will use that for the purposes of discussion and illustration. However, BIPD tables can be developed for input signals of an arbitrary number of non-commensurate signals. As an example, consider the case of a down-conversion mixer where a narrowband RF signal and a local oscillator signal are to be mixed through an intentional non-linearity that is known to have non-linear behavior up to the third order. More formally, the input is a signal of the form $x(t) = X_1 \cos\omega_1 t + X_2 \cos\omega_2 t$, where we are assuming that $\omega_2 > \omega_1$, and the mixer has a non-linear behavior that can modeled as $y = a_0 + a_1 x + a_2 x^2 + a_3 x^3$. Then because of the third-order non-linearity, we can expect to see harmonic distortion products at $3\omega_1$ and $3\omega_2$ as well as intermodulation distortion products at $2\omega_1 - \omega_2$ and $2\omega_2 - \omega_1$. In a down-conversion mixer, we are not generally concerned with third-order harmonics, since they can be filtered easily through a bandpass filter. However, the intermodulation distortion products just mentioned are of great concern because they surround the two fundamental frequencies of interest, ω_1 and ω_2, and often fall within the passband of filters for the output y. Since they are of great concern to us, we will wish to use AOM to identify their possible effect on the output signal. The second order non-linearity will yield second order harmonics at $2\omega_1$ and $2\omega_2$ along with intermodulation products at $\omega_2 - \omega_1$ (the desired down-conversion result) and $\omega_2 + \omega_1$. We have thus identified 9 output frequencies in this example in which we are interested: 0 (DC), $\omega_1, \omega_2, \omega_2 - \omega_1, \omega_2 + \omega_1, 2\omega_1 - \omega_2, 2\omega_2 - \omega_1, 2\omega_1$, and $2\omega_2$. These are the Basic Intermodulation Product Description frequencies of interest to us. Thus, the Spectrum Transform Matrix we create for the problem should have a set of two rows and columns corresponding to each frequency. As noted in the paper by Chang and Steer, since the DC term of the output will have no

imaginary part, we simply delete the second row and column from the matrix after executing the algorithm to create it. Thus for the case of DC and 8 non-zero frequencies, the Spectrum Transform Matrix will be 17x17.

We can describe the BIPD frequencies by means of a table that describes the relationships among the frequencies using a vector algebra form. If f_1 and f_2 are the two non-commensurate frequencies that comprise the input signal $x(t)$, i.e. $\omega_1 = 2\pi f_1$ and $\omega_2 = 2\pi f_2$, then we can describe the BIPD frequencies by means of the following algebraic weighting table:

BIPD Table

kx	n_1	n_2
0	0	0
1	1	0
2	0	1
3	-1	1
4	1	1
5	2	-1
6	-1	2
7	2	0
8	0	2

Here, k_x is a mnemonic index that we use to distinguish different frequencies. By convention, we denote DC by $k_x = 0$ and assign the non-commensurate tones to the values of k_x corresponding to the description we gave for $x(t)$ above. With this construction, we can now assign the non-commensurate input tones a unit vector algebraic description in terms of the frequency weighting factors n_1 and n_2. Doing so allows us to describe the harmonic and intermodulation products as weighted sums of the non-commensurate input frequencies. Note, however, that beyond DC and the non-commensurate input frequencies, there is no functional relationship between k_x and the harmonic or intermodulation tone it describes. The index is simply a tracking mechanism for these tones. With this table in hand, we now know the domain of the spectrum of the output – one of the prerequisites at the outset. That is, we now know the size of the Spectrum Transform Matrix $\mathbf{T}_x$. However, we have not yet determined the range of the output, which is to say, the contents of the Spectrum Transform Matrix. This comes from the Spectrum Mapping Table, to which we now turn our attention.

2.6.2 Spectrum Mapping Table

The rationale for describing the BIPD frequencies in vector form becomes apparent when we confront the prospect of trying to determine the output at each of the frequencies in the table. Consider again the polynomial transfer function of the output for our example: $y = a_0 + a_1 x + a_2 x^2 + a_3 x^3$. Since in the frequency domain this polynomial corresponds to repeated convolution operations, we must allow for the possibility that any of the frequencies in the BIPD table – including the harmonic and intermodulation products – will be further mixed within the mixing apparatus to produce additional phasor components at each BIPD frequency. We must specifically be alert to look for instances of any entry in the BIPD table being subtracted from any other entry where the resulting frequency maps back onto one of the BIPD frequencies. We must determine every instance where this does occur, because the output y at each frequency in the BIPD table will be composed of a sum of phasor products of exactly this sort. For

example, consider the result of subtracting the tone at frequency index $k_x = 3$ (produced from the second-order mixing) from the tone at frequency index $k_x = 6$, (produced from third-order intermodulation):

Spectrum Mapping

kx	n_1	n_2
-3	1	-1
6	-1	2
Sum	0	1

The result of this additional frequency convolution is a phasor product term that maps to the same output frequency as the non-commensurate fundamental tone f_2. There may be several phasor component products generated in this fashion that map onto the output BIPD frequencies. The enumeration of these frequencies forms the Spectrum Mapping Table. Unlike the simple illustrative example here, the Spectrum Mapping Table is constructed systematically by fixing the output BIPD frequency and then finding all vector combinations of the input BIPD frequencies – including vector subtraction combinations – that map to the output BIPD frequency. Each entry in the table will contain 5 pieces of information: The frequency index k_y of the output BIPD to which the phasor product is being mapped, the frequency indices of the two input BIPDs being mixed, and the *signs* indicating frequency addition or subtraction. For example, if we denote our two input signals using frequency indices k_x and k_z and denote the signs of the vector operation as s_x and s_z, then the entry in the Spectrum Mapping Table for the previous example would be:

Spectrum Mapping Table Entry

ky	kx	sx	kz	sz
2	3	-1	6	1

It must be emphasized that the Spectrum Mapping Table can, and often does, contain more than one entry with the same value for k_y, this being the result of many different forms of intermodulation taking place among the BIPD frequencies. If we now refer back to equation (1.20), we can formulate the phasor result of the operation specified in the Spectrum Mapping Table Entry in terms of the BIPD nomenclature:

$$Y'_{k_y} = \left(\varepsilon X_{k_x r} + j\varepsilon s_x X_{k_x i}\right)\cdot\left(Z_{k_z r} + j s_z Z_{k_z i}\right) = \varepsilon X_{k_x r} Z_{k_z r} - s_x s_z \varepsilon X_{k_x i} Z_{k_z i} + j\left(s_x \varepsilon X_{k_x i} Z_{k_z r} + s_z \varepsilon X_{k_x r} Z_{k_z i}\right) \tag{1.26}$$

In the spirit of equations (1.21) through (1.23), we can restate the basic formulation of the Spectrum Transform Matrix entries using the result in equation (1.26). We will have:

$$\begin{bmatrix} Y'_{k_y r} \\ Y'_{k_y i} \end{bmatrix} = \begin{bmatrix} \varepsilon X_{k_x r} & -\varepsilon s_x s_z X_{k_x i} \\ \varepsilon s_x X_{k_x i} & \varepsilon_z X_{k_x r} \end{bmatrix} \begin{bmatrix} Z_{k_z r} \\ Z_{k_z i} \end{bmatrix} \tag{1.27}$$

It is important to note the primes on Y in equations (1.26) and (1.27) and emphasize that we have shown the result of the inclusion of only one BIPD mixing product here. As the paper by Chang and Steer [2] notes, the construction of the Spectrum Transform Matrix is done by first initializing all entries to zero and then adding to the matrix based upon the specifications of the Spectrum Mapping Table. Once all of the entries in the Spectrum Mapping Table have been read and updates made to the Spectrum Transform Matrix, the Spectrum Transform Matrix $\mathbf{T}_x$ is now complete. To begin using $\mathbf{T}_x$ to perform frequency

domain convolution operations, all that remains is to construct the spectral vector, $\mathbf{Z}$ or $\mathbf{X}$ (which are the same in the example in this section, although we used the $\mathbf{Z}$ variable for clarity in explanations). $\mathbf{Z}$ must have the same number of rows as $\mathbf{T}_x$, which in our example is 17. The construction is straightforward: Given that $x(t) = X_1 \cos\omega_1 t + X_2 \cos\omega_2 t$, $\mathbf{Z}$ is a vector containing a zero entry (for DC) followed by the phasor values for its two non-commensurate tones, followed by a set of padding zeros corresponding to the harmonic and intermodulation terms. Thus we have:

$$\mathbf{Z} = [0, X_{1r}, X_{1i}, X_{2r}, X_{2i}, 0,0,0,0,0,0,0,0,0,0,0,0]^{\mathrm{T}} \tag{1.28}$$

Once the series of matrix multiplications corresponding to each polynomial term has been carried out as called for by the form for y in (1.25), the result is a vector of phasor coefficients for each polynomial term in y. Thus the time domain $y(t)$ can be recovered by simply converting the phasor coefficients for each polynomial product into amplitude and phase terms and forming the cosine function with frequency $2\pi f_{k_y}$ as determined by the BIPD table, and then summing the cosines that correspond to each polynomial product.

3. AOM Case Study – A Logarithmic Amplifier

Consider the logarithmic amplifier shown in Figure 1. Such circuits are most often used in instrumentation systems because of the wide range (many orders of magnitude) of inputs to which they can respond (see, for example, [9] and [10]). Its transfer function can be written as an infinite series expansion relating the output $y(t)$ to the input $x(t)$. The Arithmetic Operator method is used to solve the expansion in the frequency domain.

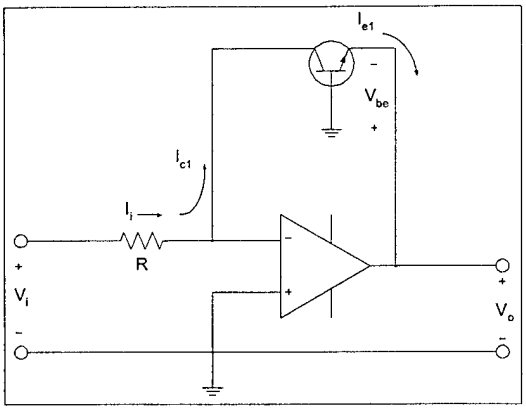

Figure 1: Logarithmic Amplifier

The following assumptions are made about the circuit elements:

1. Ideal Bipolar Transistor
2. Ideal op-amp: The voltage between terminals tends to zero, and the current into the op-amp is zero. $V_o = -V_{be}$, and $I_{c1} = I_i$.

From the transistor constitutive relations:

$$I_{c1} = I_s\left[\exp(\tfrac{V_{be}}{V_T}) - 1\right] \approx I_s \exp(\tfrac{V_{be}}{V_T}) \qquad (1.29)$$

where I_s is the reverse saturation current of the transistor, and V_T is the thermal voltage. Rearranging terms,

$$\tfrac{I_{c1}}{I_s} = \exp(\tfrac{V_{be}}{V_T}) \qquad (1.30)$$

$$\ln(\tfrac{I_{c1}}{I_s}) = \tfrac{V_{be}}{V_T} \qquad (1.31)$$

$$V_{be} = V_T \ln(\tfrac{I_{c1}}{I_s}) \qquad (1.32)$$

$$V_o = -V_T \ln(\tfrac{I_{c1}}{I_s}) \qquad (1.33)$$

Now, $I_i = I_{c1} = \tfrac{V_i}{R}$, so

$$V_o = -V_T \ln(\tfrac{V_i}{RI_s}) \qquad (1.34)$$

For use with the Arithmetic Operator Method, this transfer characteristic should be represented in terms of the basic mathematical operators - addition, subtraction, multiplication, and division. Three common forms exist for expanding the natural logarithm [11]:

$$\ln(x) = (x-1) - \tfrac{1}{2}(x-1)^2 + \tfrac{1}{3}(x-1)^3 + \ldots \quad (0 < x \le 2) \qquad (1.35)$$

$$\ln(x+1) = x - \tfrac{1}{2}x^2 + \tfrac{1}{3}x^3 - \tfrac{1}{4}x^4 + \ldots \quad (-1 < x \le 1) \qquad (1.36)$$

$$\ln(x) = 2[\tfrac{x-1}{x+1} + \tfrac{1}{3}(\tfrac{x-1}{x+1})^3 + \tfrac{1}{5}(\tfrac{x-1}{x+1})^5 + \tfrac{1}{7}(\tfrac{x-1}{x+1})^7 + \ldots \quad (x > 0) \qquad (1.37)$$

For the purposes of this paper, the circuit equation was manipulated to fit the form of (1.36). The input can be written as

$$V_i = x_{dc} + x_{ac} = x_{dc}(1 + \tfrac{x_{ac}}{x_{dc}}), \text{ where } \left|\tfrac{x_{ac}}{x_{dc}}\right| < 1 \qquad (1.38)$$

Substituting (1.38) into (1.34) results in

$$V_o = -V_T \ln\left[\frac{x_{dc}(1+\frac{x_{ac}}{x_{dc}})}{RI_s}\right], \text{ and this can be expanded as:} \qquad (1.39)$$

$$V_o = -V_T\left[\ln(x_{dc}) + \ln(\tfrac{1}{RI_s}) + \ln(1 + \tfrac{x_{ac}}{x_{dc}})\right] \qquad (1.40)$$

Of the terms in (1.40) the first two are constant offsets. Only the last term has frequency-dependent characteristics. This last term can be represented by its linear series expansion shown in (1.36) and solved with the Arithmetic Operator Method.

The comparison was performed for an input signal consisting of the sum of two sinusoids:

$V_i = A_0 + A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t)$, which in the form of (1.38) is:

$$V_i = A_0[1 + \frac{A_1}{A_0}\sin(2\pi f_1 t) + \frac{A_2}{A_0}\sin(2\pi f_2 t)] \qquad (1.41)$$

The sine form was chosen in order to assure convergence at the beginning of a transient simulation in Spice. Since sine and cosine are shifted 90 degrees in phase, that is, $\sin\omega t = \cos(\omega t - 90°)$, we take the phase into account by designating the amplitudes of the non-commensurate input tones above in the spectral vector as pure imaginary terms. Parameter values are shown in the Table 1:

Circuit and Input Parameters

Parameter	Value	Unit
A_0	10	V
A_1	2	V
A_2	2	V
f_1	0.9	GHz
f_2	1	GHz
R	1,000	Ohm
I_s	1.00E-16	Amp

Table 1: Circuit And Input Parameters.

The Arithmetic Operator Method requires the input signal to be a spectral vector of the form

$\mathbf{X} = [X_0\ X_{1r}\ X_{1i}\ X_{2r}\ X_{2i} \ldots X_{kr}\ X_{ki}]^T$, where

- X_0 is the dc term,
- k is number of frequencies in the BIPD table, and
- X_{kr},X_{ki} represents the complex number pair (real part and imaginary) describing the amplitude and phase of the kth frequency component.

To reach this form, rewrite (1.41) as

$$V_i = A_0[1+x], \quad \text{where} \quad x = \frac{A_1}{A_0}\sin(2\pi f_1 t) + \frac{A_2}{A_0}\sin(2\pi f_2 t) \qquad (1.42)$$

Now, for the input signal in (1.42), the AOM input is the frequency domain spectral vector of x, $\mathbf{X}$, and consists of the following components:

$X_0 = 0$
$X_{1r} = 0$
$X_{1i} = -A_1/A_0$
$X_{2r} = 0;$
$X_{2i} = -A_2/A_0$

As will be explained further in just a moment, there are 10 frequencies in the BIPD table, so the spectral vector $\mathbf{X}$ must contain 2*10+1 = 21 terms. Thus, to form the spectral vector $\mathbf{X}$ simply pad the input vector with zeros to obtain the correct size. The spectral vector for this simulation is:

$$\mathbf{X} = [0,0,-(A_1/A_0),0,-(A_2/A_0),0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]^T \qquad (1.43)$$

Now we must create the Spectrum Transform Matrix $\mathbf{T_x}$. To do this, we need to choose a set of BIPD frequencies. For this example, we are interested in all of the frequencies in the frequency domain surrounding the fundamentals and its second harmonics. We could choose to have more frequencies, but since we can sometimes put second harmonics to good use [12], while third harmonic content is of little use and can be filtered, we will choose the second harmonic content limit. This gives us the following BIPD table:

BIPD Table		
kx	n_1	n_2
1	1	0
2	0	1
3	-1	1
4	2	-1
5	-1	2
6	2	0
7	0	2
8	3	-1
9	1	1
10	-1	3

Table 2: BIPD Table.

Notice in Table 2 that we are not ignoring third order intermodulation products, as the indices for $k_x = 4$ and $k_x = 5$ show. Furthermore, there will be some fourth order intermodulation products that appear in the range of the second harmonics, so we've added the frequency indices $k_x = 8$ and $k_x = 10$ in order to take account of this. To gain some visual insight into the set of frequencies we are considering, the Table 2 above is plotted graphically in Figure 2 below using our selected values for the non-commensurate tones, $f_1 = 0.9$ GHz and $f_2 = 1.0$ GHz. (Note: The plot shown in Figure 2 accurately shows the locations of the BIPD frequencies, but does not give an accurate depiction of the output amplitudes we can expect from the logarithmic amplifier. This is because the plot was done assuming an arbitrary transfer function of the form $y(t) = x(t) + x^2(t) + x^3(t) + x^4(t)$ with $x(t) = 1\cos(2\pi f_1 t) + 2\cos(2\pi f_2 t)$ used as the input, and where some of the fourth order mixing components have been intentionally dropped.)

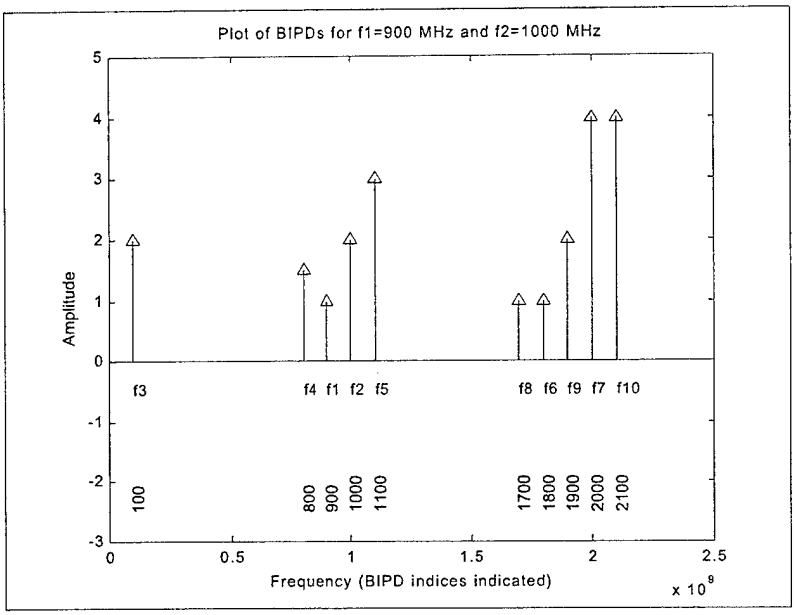

Figure 2: BIPD Locations.

Comparing the table above with that shown earlier in the 2.6.1 Basic Intermodulation Product Description Table section, we do not carry DC as a BIPD in the Table 2. We did not overlook DC in our AOM modeling, but we have found that it is more convenient to take DC mixing terms into account at the time the Spectrum Mapping Table is constructed when using computer-aided methods, and this is what we have done. This is because the DC terms created through mixing of BIPD entries are essentially composed of fundamental inputs (designated by $k_x = 1$ and $k_x = 2$ in the table above) mixing with themselves through subtractive intermodulation. Using the systematic methods described previously, the Spectrum Mapping Table is created, and from this the Spectrum Transform Matrix. For the example here with 10 non-zero BIPD terms, the Spectrum Mapping Table has 115 entries. With the Spectral Vector and the Spectrum Transform Matrix now defined, we, perform the series expansion of (1.36) in the frequency domain for the output spectral vector **Y**:

$$\mathbf{Y} = \mathbf{X} - \frac{1}{2}\mathbf{T}_x\mathbf{X} + \frac{1}{3}\mathbf{T}_x(\mathbf{T}_x\mathbf{X}) - \frac{1}{4}\mathbf{T}_x(\mathbf{T}_x(\mathbf{T}_x\mathbf{X})) + \ldots \qquad (1.44)$$

For this simulation, the infinite series expansion as described in equation (1.36) was truncated at 20 terms. Figure 3 below shows the accuracy of the 20 terms approximation over the interval $(-1 < x \leq 1)$. Note that with the magnitude x of the input having a maximum value of 0.4, we can expect very little error in the results attributable to the decision to truncate the series expansion at 20 terms. But since the number of terms has a direct impact on simulation time, minimizing the number of terms given an accuracy target should always be a goal. We established our accuracy target through graphical inspection of Figure 2, but it is feasible to dynamically adapt the number of terms depending on the iteration error. This is a convenient place to optimize a simulation beyond what is presented here.

Transforming **Y** back to the time domain, and adding the constant terms of (1.40), the AOM simulation results compare favorably with Spice output as shown in Figure 4.

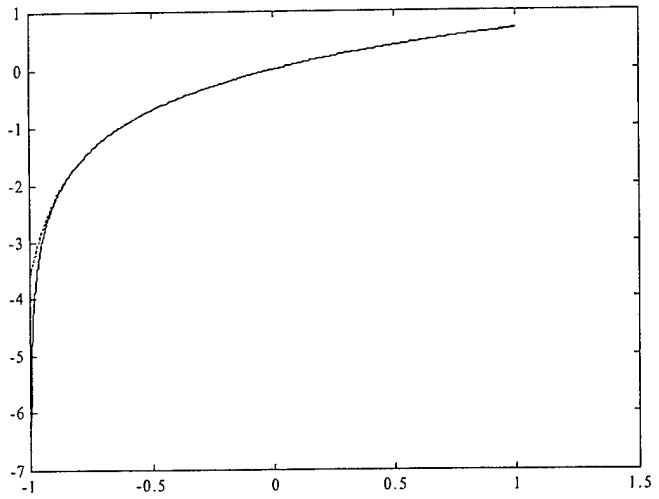

Figure 3: Series Expansion of ln(1+x) for 20 terms. Dotted line is predicted; solid is actual. X-axis is x; Y-axis is ln(1+x).

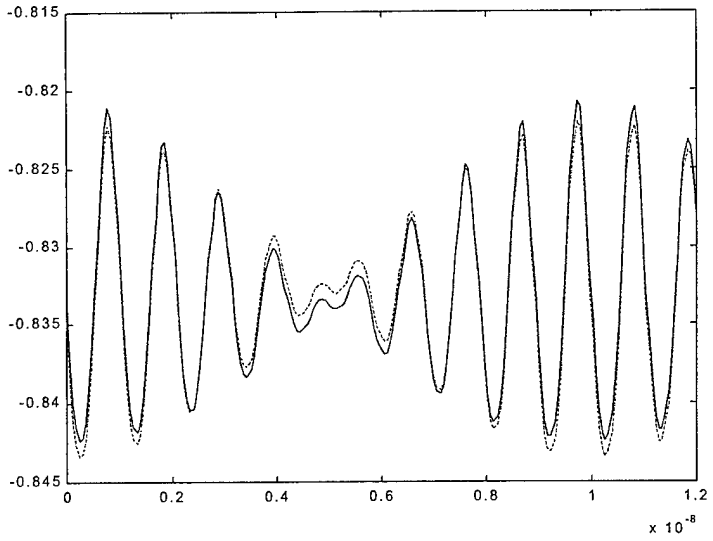

Figure 4: Logarithmic Amplifier Simulation: AOM (dotted line) compared with Spice (solid line). X-axis is time; Y-axis is V_O.

The results indicate good agreement with the Spice output. The small discrepancies that one sees can be attributed to two factors: First, the Spice model used a gain of 100 for the operational amplifier rather than an infinite value, so the Spice output itself may have some error compared to a completely ideal model. Running Spice with the op amp gain as a parametric value would make for an interesting area of further research. Second, since we have demonstrated that a 20 term polynomial expansion for the natural logarithm function yields an accurate approximation given the magnitude limits on x, the error is likely attributable to deviations in the Spectrum Transform Matrix. We get deviations in the Spectrum Transform Matrix through our choice of BIPD frequencies. It is likely that increasing the number of BIPD frequencies will lead to more accurate results. Referring back to Table 2, we chose to limit our fourth order intermodulation terms to only two – those for $k_x = 8$ and $k_x = 10$. We will, however, also have fourth order intermodulation products at DC due to subtractive mixing of the second harmonic frequencies, these were not taken into account, and it is possible that their inclusion could increase the accuracy of the results.

4. Conclusion

We have explained the Arithmetic Operator Method (AOM) mathematics in some detail and shown the relationship between the mathematics and the algorithms used to implement the method in the paper by Chang and Steer [2]. We have also presented a case study of the application of AOM for a logarithmic amplifier and demonstrated that the results of using the method compare favorably to those from a Spice reference simulation model. It is our view that these favorable results confirm the utility of the method for behavioral modeling of devices, and it is our hope that our explanations and case study will help others apply AOM.

References

1. C.R. Chang, "Computed-Aided Analysis of Nonlinear Microwave Analog Circuits Using Frequency Domain Spectral Balance," Ph.D. thesis, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 1993, pages 36-49.
2. C.R. Chang and M.B. Steer, "Frequency-Domain Nonlinear Microwave Circuit Simulation Using the Arithmetic Operator Method," *IEEE Transactions on Microwave Theory and Techniques*, Volume 38, Number 8, August, 1990, pages 1139-1143.
3. K.G. Gard, H.M. Gutierrez, and M.B. Steer, "Characterization of Spectral Regrowth in Microwave Amplifiers Based on the Nonlinear Transformation of a Complex Gaussian Process", *IEEE Transactions on Microwave Theory and Techniques*, Volume 47, Number 7, July 1999, pages 1059-1069.
4. N.B. de Carvalho and J.C. Pedro, "Simulating Strong Non-Linear Microwave Circuits Driven By a Large Number of Input Tones," 27th European Microwave Conference, Jerusalem, September, 1997.
5. N.B. de Carvalho and J.C. Pedro, "Simulation of Multi-Tone IMD Distortion And Spectral Regrowth Using Spectral Balance," *IEEE Microwave Theory and Techniques Digest*, 1998.
6. A.E. Taylor and W. R. Mann, *Advanced Calculus, 3ed*, Wiley, 1983, pages 650-652.
7. A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing*, Prentice-Hall, 1975, pages 480-500.
8. W.E. Boyce and R.C. DiPrima, *Elementary Differential Equations and Boundary Value Problems, 3ed*, Wiley, 1977, pages 257-260.
9. G. Acciari, F. Giannini, and E. Limiti, "Theory and performance of parabolic true logarithmic amplifier," *IEEE Proceedings – Circuits, Devices, and Systems*, Volume 144, Number 4, August 1997, pages 223-228.
10. W.G. Jung, *IC Op Amp Cookbook, 3ed*, Sams, 1986, pages 290-292.
11. W.H. Byer, ed., *CRC Standard Mathematical Tables, 26th Ed.*, 1981, CRC Press, page 349.
12. M.B. Steer and C.E. Christofferson, *Circuit Simulation*, North Carolina State University pre-publication manuscript, Jan. 10, 2001, Chapter 6.